Solution Sample paper- 10

Q.1	(iii) Pea Cock # Blank Space not allowed			
Q.2	(i) if keyword			
	(ii) roll_no identifier / variable			
Q.3	vemyCou			
Q.4	300 # 100			
	300 # 200			
	240 # 200			
Q.5	b)Move file pointer five character ahead from the beginn	ing of a file		
Q.6	b)writelines(list)			
Q.7	d) Ture			
	Ture			
Q.8	c) Error # dictionaries are un-ordered type, so can't com	npare the dictionary.		
Q.9	a)dict.keys()			
Q.10	a)Create an instance of a cursor			
Q.11	a) Super Key b) Candidate Key c) Primary Key d) All of the	above Answer: d) All of the above		
Q.12	b) Flase			
Q.13	d) DISTINCT			
Q.14	c) IS			
Q.15	b) ghagwal			
Q.16	d) SMTP			
Q.17	a) Both A and R are true and R is the correct explanation f	for A		
Q.18	c) A is True but R is False			
Q.19	x=int(input("Enter a number ")) # int() r	missing, and wrong variable 5x		
	<u>if(abs(x)==x</u>): # shoul	d == operator for comparison		
	print ("You Entered a positive number")			
	else: # else:	should in lower case letters		
	$\mathbf{x} = \mathbf{x}^{+}(-1)$	and the life of the second second		
0.00	print("Number made positive <u>: ",x</u>) # string	and x should separate with comma		
Q.20	default parameter(s) These are the part of function definition	keyword parameter(s)		
	The name of both actual and formal narameters may	The name of both actual and formal parameters are		
	he different	he same		
	The order of parameters cannot change	The order of actual parameter can change.		
	When no actual parameter passed then the default	When no actual parameter passed then error will be		
	When no actual parameter passed then the default value take place of formal parameter and no error will	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory.		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory.		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required.		
	 When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. 	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required.		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example:	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a b);		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b):	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b):		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b): print(a " is big")	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b): print(a, " is big")		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b): print(a, " is big") else:	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b): print(a, " is big") else:		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b): print(a, " is big") else: print(b, " is big")	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b): print(a, " is big") else: print(b, " is big")		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b): print(a, " is big") else: print(b, " is big") big(10,15)	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b): print(a, " is big") else: print(b, " is big") big(a=10,b=15)		
	When no actual parameter passed then the default value take place of formal parameter and no error will be occurred The sequence of setting default parameter is from right most side to left side in continuous way. Example: def big(a,b=0): if(a>=b): print(a, " is big") else: print(b, " is big") big(10,15) big(12)	When no actual parameter passed then error will be occurred. Mean all parameter values are mandatory. No such order required. Example: def big(a,b): if(a>=b): print(a, " is big") else: print(b, " is big") big(a=10,b=15) big(b=12,a=14)		

	Local (Private) Variable	Global (Public) Variable	
	Local variable are declared inside of function or block	Local variable are declared inside of function or block	
	Local variable can access within the function where it	Local variable can access within the function where it	
	was declared	was declared	
	Visibility of Local variable is limited to function where	Visibility of Local variable is limited to function where	
	it was declared, hence cannot access outside the	it was declared, hence cannot access outside the	
	function	function	
	If in the program there is a public variable with the	If in the program there is a public variable with the	
	same name of local variable is available then local	same name of local variable is available then local	
	variable cannot change the value of global variable.	variable cannot change the value of global variable.	
	Scope of Local variable is limited.	Scope of Local variable is limited.	
	Example:	Example:	
	def local():	pub=100 # pub is global variable	
	loc=10 # loc is local variable	def public():	
	print(loc)	print(pub) # output 100	
		public()	
	print(loc) # loc is not accessible here	print(pub) # output 100	
Q.21	(a) \$\$\$\$\$Sequence with labels\$\$\$\$		
	(b) ('a'. '3'. 'sum'. '4')		
0.22	a) ORDER BY		
Q.22	* It is the clause that used to arrange the records of SI	ELECT query in the desired order either in ASCENDING or	
	DESCENDING order.		
	* It can works on one or more Attributes of Table.		
	* The default order is ASCENDING and command is ASC, v	where for DESCENDING order command is DESC.	
	* The ORDER BY clause will be the last clause (statement)	in any SQL query.	
	* Example (Display records of students in descending or	ler of their names)	
	SELECT * FROM student		
	ORDER BY name DESC;		
	b) HAVING		
	* It is the clause that used for making criteria (condition)	in SQL query.	
	* It always used only with GROUP BY clause for defining condition.		
	* Example (Display records of students in the group of their cities, whose marks are greater than 80)		
	SELECT * FROM student		
	GROUP BY city		
	HAVING marks > 80 ;		
Q.23	a) IMAP : Internet Message Access Protocol		
0.24			
Q.24	kVS^NISHANT^3134		
Q.25	Maximum Value for Variable Y = 4		
	Minimum Value for Variable Y = 0		
	Output:(i) 0 : 0(iv) 0 : 3Both are correct		
Q.26	(i) MAX (PRICE) MIN(QTY_STORE)		
	15000 100		
	1 49		
	2 19		
	3 12		

	(iii) <u>watch_name</u>	QTY_STOP	RE SUM (QTY_	<u>SOLD)</u>
	High Time	100	22	
	Life Time	150	23	
	Wave	200	26	
	High Fashion	135	9	
Q.27	def COPY():			
	fread=open("TEXT1	1.TXT",'rt')		
	fwrite=open("TEXT	2.TXT",'wt')		
	data=fread.read()			
	words=data.split()			
	ior with words:			
	furito(m)			
	fread close()			
	fwrite close()			
	OR			
	def countmy():			
	f=open("data.txt","	rt')		
	rec=f.read()			
	words=rec.split()			
	count=0			
	for w in words:			
	if(w =="my"):			
	count+=1			
	print("No. of time r	my word foun	d in file: ",count)	
0.29	1.close()			
Q.28	(a) <u>COUNT(*)</u>	VCODE		
	2 V	/01		
	$(D) \underline{VO2} \underline{CODE} \underline{C}$	avi Anish		
	V02 In	ohn Malina	CAR	
	(c) DISTINCT VCODE			
	V01			
	V02			
	V02 V02			
	V03			
	V05			
	V05	(-)		
Q.29	def count_UP_low((S):		
	up=low=0			
	for ch in S:			
	if(ch.isupper())	:		
	up+=1			
	elif(cn.islower()):		
	IOW+=1			
	print("Original Sti	ring: ",S)	· · · · · · · · · · · · · · · · · · ·	
	print("Upper case	e letter in Str	ing: ",up)	
	print("Lower case	e letter in Str	ing: ",iow)	
	count UP low/"Pvt	hon Program	nmiNg")	
0.30	def PushOn/Studen	t).	ъ /	
Q.30		· /·		

	name=input("Enter Name of Student to push in stack: ")
	Student.append(name)
	def Pop(Student):
	if(len(Student)>0):
	Student.pop()
	else:
	print("Stack Underflow")
	PushOn(['Ajay', 'Vijay'])
0.21	POP([AJdy , VIJdy])
Q.31	Head Office and Tech Office: LAN (Distance is 70 Meter in between both offices)
	Head Office and Coimbatore Office: WAN (Distance is 1950 KM in between both offices)
	b) (i) Switch/Hub
	c) (ii) Optical fibre (For very effective and fast communication)
	d) a Cable /Wiring layout for connecting the company's local offices located in New Delhi
	Head Office
	10 Mtr
	SalaeOffice
	Tech Office
	an effective method /technology for connecting the company 's regional office at "Kolkata","Coimbatore and
	"Ahmedabad"
	Answer: Satellite communication, Because the distance among offices is very much in kilometers.
Q.32	def PushEl(element):
	name=input("Enter Name of element to push in stack: ")
	element.append(name)
	def Make El(alement)
	if(lon(clement)>0);
	element pop()
	else
	print("Stack Underflow")
	PushEl(['Computer', 'Keyboard'])
	MakeEl(['Keyboard'])
	OR
	def Push(A):
	stack=[]
	print("Original List: ".A)
	for n in A
	if(n%2==0)
	$r_{1/2} = -0$
	Stack.appenu(n)
	IT(IEN(STACK)>U):
	print(" Number in Stack: ", stack)
	else:

	print("No Number found in stack")			
	Push([12,31,43,54,57])			
Q.33	def countVowels():			
	f=open("dmyfile.txt",'rt')			
	rec=f.read()			
	count=0			
	for ch in rec:			
	if(ch in "aeiouAEIOU"):			
	count+=1			
	print("Count of vowels in file: ",count)			
	f.close()			
Q.34	(i) <u>M Company</u> <u>M Name</u> <u>M Pric</u> e			
	Nokia N1100 2250			
	Sony XperiaM 7500			
	Samsung Galaxy 4500 Micromax Unite3 4500			
	Oppo SelfieEx 8500			
	(ii) <u>M_Id</u> <u>M_Company</u> <u>M_Name</u> <u>M_Price</u> <u>M_Mf_Date</u>			
	MB001 Samsung Galaxy 4500 23-Jan-2004			
	MB005 Sony XperiaM 7500 01-Jan-2004			
	(iii) <u>M Supplier M Qty</u>			
	New Vision 450			
	Classic Mobile Store 300			
	A-one-Mobiles 150 Mobile Centre 50			
	(iv) <u>M Company</u>			
	Samsung			
	Micromax			
	Sony			
	(v) <u>M_Company</u> M_Supplier_ <u>M_Pric</u> e			
	Samsung Classic Mobile Store 4500			
0.35	Micromax New Vision 4500			
Q.33	(a) Line-1. In point csv (b) Line 3: $(w', or (a'))$			
	(b) Line-2. W Of a (c) Line-3: $csv reader()$			
	(d) Line- A : csvEile close()			
	(e) Line-5: Output:			
	('Note Book', 45, 100)			
	('Text Book', 60, 150)			
	('Ball Pen', 10, 100)			
	('Pencil', 5, 200)			